This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A data structure generation system comprising:

a plurality of data structure components, each data structure component configured to have a precedence defining an override level of the data structure component, to include one or more embedded rules, and to include content; and

a computer-implemented knowledge base configured to be coupled to a data structure assembly facility, the knowledge base configured to store the plurality of data structure components as objects in an object-relational hierarchy, the knowledge base storing a first set of objects and a second set of objects, the first set of objects having a first, read-only precedence level and the second set of objects having a second precedence level higher than the first precedence level.

2. (Original) A system as claimed in claim 1, wherein the precedence provides hierarchical control of content to match business preferences.

3. (Canceled)

4. (Original) A system as claimed in claim 3, further comprising a data structure assembly facility.

5. (Previously Presented) A system as claimed in claim 4, wherein the assembly facility is configured to retrieve one or more data structure components from the knowledge base; process the one or more data structure components in a processor to generate a tree having a root node; process the tree beginning at the root node; and to override objects of low precedence with objects of high precedence.

6. (Previously Presented) A system as claimed in claim 5, wherein the assembly facility is configured to, when an object having a rule is encountered, evaluate the rule and replace the rule with a value.

7. (Original) A system as claimed in claim 4, further comprising an authoring tool and a content management system.

8. (Previously Presented) A system as claimed in claim 7, wherein the content management system is configured to permit a user to create a version of an object in the first set of objects, and save the version of the object at a precedence different than the first precedence level.

9. (Original) A system as claimed in claim 7, wherein the content management system is configured to permit a user to create a version of an object in the first set of objects where the version of the object and the object at a different precedence level have the same name.

10. (Original) A system as claimed in claim 1, wherein each object is configurable to be locked in order to prevent overriding by an object having a higher precedence level.

11. (Currently Amended) A computer-implemented knowledge base configured to store data structure components as objects in an object-relational hierarchy, each object configured to have a precedence defining an override level of the object, to include one or more embedded rules, and to include content, the knowledge base storing a first set of data components stored as objects and a second set of data components stored as objects, the first set of data components stored as objects having a first, read-only precedence level and the second set of data components stored as objects having a second precedence level higher than the first precedence level.

12. (Canceled)

13. (Previously Presented) A knowledge base as claimed in claim 12, wherein each data component stored as an object is configurable to be locked in order to prevent overriding by an object having a higher precedence level.

14. (Currently Amended) A computer-implemented method of assembling a data structure from a group of components, the method comprising:

retrieving one or more cross-referenced data structure components from a database, the one or more data structure components configured to have a precedence level defining an override level for the one or more data structure components;

processing the one or more cross-referenced data structure components in a processor to generate a tree having a root node;

processing the tree beginning at the root node;~~and~~

overriding objects of low precedence with objects of high precedence to create a resulting tree; and

transforming the resulting tree into a data structure representing a document.

15. (Original) A method as claimed in claim 14, further comprising creating a transaction data set.

16. (Previously Presented) A method as claimed in claim 15, wherein retrieving one or more cross-referenced data structure components from a database includes retrieving the same based on the transaction data set.

17. (Original) A method as claimed in claim 15, wherein the one or more data structure components are configured to include one or more rules.

18. (Previously Presented) A method as claimed in claim 15, further comprising, when a rule is encountered, evaluating the rule and replacing the rule with a value.

19. (Canceled)

20. (Previously Presented) A method as claimed in claim 14, further comprising configuring each data structure component to be lockable in order to prevent overriding by an object having a higher precedence level.

21. (Previously Presented) A method as claimed in claim 14, further comprising configuring the database so that it may include a first set of data structure components, a second set of data structure components, and a third set of data structure components, the first set of data structure components having a first, read-only precedence level, the second set of data structure components having a second precedence level higher than the first precedence level, and the third set of data structure components having a third precedence level higher than the second precedence level

22. (Previously Presented) A computer readable medium containing instructions for generating a data structure by

retrieving one or more cross-referenced data structure components from a computer-implemented database, each of the one or more data structure components configured to have a precedence level defining an override level for the data structure component;

processing the one or more cross-referenced data structure components in a processor to generate a tree having a root node;

processing the tree beginning at the root node;

overriding objects of low precedence with objects of high precedence to create a resulting tree; and

transforming the resulting tree into a data structure representing a document.

23. (Previously Presented) A computer readable medium as claimed in claim 22, further comprising instructions for structuring the one or more data structures to include one or more embedded rules.

24. (Previously Presented) A computer readable medium as claimed in claim 23, further comprising instructions for processing the one or more data structures components by evaluating a rule and replacing the rule with a value when the rule is encountered.